Listing of Claims

1-45. (Cancelled)

46. (Original) A method for adding tamper resistance to a software program, the method comprising:

installing a silent guard in a software program, said silent guard comprising one or more program instructions.

47. (Original) The method of claim 46, further comprising the step of:

evaluating the integrity of one or more data items in computer memory when said software program is running; and

taking a defensive action if said silent guard detects a deficiency in said integrity of said one or more data items.

48. (Original) The method of claim 46, further comprising the step of:

assigning, by operation of said silent guard, a predetermined value to a software program variable in computer memory when said software program is running before said software program variable is used in a computation.

49. (Original) The method of claim 46, wherein said silent guard comprises a variable whose value is computed during execution of said software program, and wherein a defensive action results if said silent guard detects an unexpected value of said variable.

50. (Original) The method of claim 49, wherein said variable has an expected value, the method further comprising the step of:

comparing a runtime value of said variable against said expected value.

51. (Original) The method of claim 50, wherein the step of comparing a runtime value of said variable against said expected value comprises the step of:

inserting one or more mathematical expressions into one or more program instructions in said software program, said one or more mathematical expressions comprising said runtime value of said variable and said expected value of said variable, wherein correct execution of said one or more program instructions depends on said runtime value of said variable being the same as said expected value of said variable.

52. (Original) The method of claim 50, wherein the step of comparing a runtime value of said variable against said expected value comprises the step of:

comparing said expected value after said expected value is processed through an algorithm with said runtime value after said runtime value is processed through said algorithm.

53. (Original) The method of claim 46, wherein said value of said variable changes at least once during program execution, the method further comprising the step of:

determining a runtime value of said variable at a point in software program execution; and

comparing said runtime value of said variable at said point in software program execution against an expected value of said variable at said point in

software program execution.

54-55. (Cancelled)

56. (Original) A recordable computer readable media having a tamper resistant software program recorded thereon, comprising:

a software program comprising one or more program instructions; and

a silent guard comprising one or more guard program instructions, said one or more guard program instructions installed in said software program.

57. (Original) The recordable computer readable media having a tamper resistant software program recorded thereon of claim 56, wherein said one or more guard program instructions are operable to evaluate the integrity of one or more data items in computer memory when said software program is running, and to take a defensive action if a deficiency in said integrity of said one or more data items is detected.

58. (Original) The recordable computer readable media having a tamper resistant software program recorded thereon of claim 56, comprising:

at least one software program variable, said one or more guard program instructions being operable to assign a predetermined value to at least one of said at least one software program variables in computer memory before said software program variable is used in a computation during execution of said software program.

59. (Original) The recordable computer readable media having a tamper resistant software program recorded thereon of claim 56, wherein said software program

comprises a variable whose value is computed during execution of said software program, and wherein a defensive action results if an unexpected value of said variable is detected.

60. (Original) The recordable computer readable media having a tamper resistant software program recorded thereon of claim 59, wherein said variable has an expected value, and wherein said defensive action results if a runtime value of said variable is not the same as said expected value.

61. (Original) The recordable computer readable media having a tamper resistant software program recorded thereon of claim 59, further comprising:

one or more mathematical expressions inserted into one or more program instructions in said software program, said one or more mathematical expressions comprising said runtime value of said variable and said expected value of said variable, wherein correct execution of said one or more program instructions depends on said runtime value of said variable being the same as said expected value of said variable.

62. (Original) The recordable computer readable media having a tamper resistant software program recorded thereon of claim 59, further comprising:

an algorithm, wherein said defensive action results if a runtime value of said variable after said runtime value is processed through said algorithm is not the same as said expected value after said expected value is processed through said algorithm.

63-66. (Cancelled)